



Calhoun: The NPS Institutional Archive

Theses and Dissertations

Thesis Collection

2008-03

Using a competitive approach to improve military simulation artificial intelligence design

Stoykov, Sevdalin

Monterey, California. Naval Postgraduate School

<http://hdl.handle.net/10945/4150>



Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>



NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

**USING A COMPETITIVE APPROACH TO IMPROVE
MILITARY SIMULATION ARTIFICIAL INTELLIGENCE
DESIGN**

by

Sevdalin Stoykov

March 2008

Thesis Advisor:
Second Reader:

Christian Darken
Curtis Blais

**This thesis written in cooperation with the MOVES Institute
Approved for public release; distribution is unlimited**

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 2008	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE Using a Competitive Approach To Improve Military Simulation Artificial Intelligence Design			5. FUNDING NUMBERS	
6. AUTHOR(S) Sevdalin Stoykov				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) The research presented in this thesis attempts to show how using a competitive approach to artificial intelligence (AI) design can lead to improvement of the AI solutions used in military simulations. To demonstrate the potential of the competitive approach, ORTS, a real-time strategy game engine, and its competition setup are used. To justify the thesis statement, a set of exploratory experiments are conducted. The experiments represent a tournament of virtual battles between base case AIs and test case AIs. The existing OTRS clients are used as base cases, and the test cases are evolved using the competitive approach to AI design described in this work. The analysis of the results from the tournament proves the advantages of the competitive approach. At the end of the thesis, some conclusions and recommendations for future work are made.				
14. SUBJECT TERMS Artificial Intelligence, Virtual Environment, Combat Modeling			15. NUMBER OF PAGES 67	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

**USING A COMPETITIVE APPROACH TO IMPROVE MILITARY
SIMULATION ARTIFICIAL INTELLIGENCE DESIGN**

Sevdalin H. Stoykov
Major, Bulgarian Army
"P. Volov" Artillery and Air Defense Academy, 1993

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN MODELING, VIRTUAL ENVIRONMENTS, AND
SIMULATION (MOVES)**

from the

**NAVAL POSTGRADUATE SCHOOL
March 2008**

Author: Sevdalin Stoykov

Approved by: Christian Darken
Thesis Advisor

Curtis Blais
Second Reader

Mathias Kolsch
Chair, MOVES Academic Committee

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The research presented in this thesis attempts to show how using a competitive approach to artificial intelligence (AI) design can lead to improvement of the AI solutions used in military simulations. To demonstrate the potential of the competitive approach, OTRS, a real-time strategy game engine, and its competition setup are used. To justify the thesis statement, a set of exploratory experiments are conducted. The experiments represent a tournament of virtual battles between base case AIs and test case AIs. The existing OTRS clients are used as base cases, and the test cases are evolved using the competitive approach to AI design described in this work. The analysis of the results from the tournament proves the advantages of the competitive approach. At the end of the thesis, some conclusions and recommendations for future work are made.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	MOTIVATION	1
B.	RESEARCH QUESTIONS.....	2
C.	SCOPE	3
D.	THESIS ORGANIZATION.....	3
II.	BACKGROUND	5
A.	APPLICATION OF AI DESIGNS IN MILITARY SIMULATIONS	5
1.	Traditional Approaches in Military Simulations.....	5
a.	<i>Lanchester’s Formulation of Combat.....</i>	<i>5</i>
b.	<i>Enrichments of Lanchester’s Models</i>	<i>7</i>
2.	AI Based Approach in Military Simulations.....	8
a.	<i>Necessity of New Paradigm in Combat Modeling</i>	<i>8</i>
b.	<i>Ilachinski’s Combat Models</i>	<i>9</i>
c.	<i>Advantages of AI in Agent-Based Combat Models</i>	<i>10</i>
3.	Military Simulations Implementing AI Solutions.....	11
a.	<i>MANA.....</i>	<i>11</i>
b.	<i>Pythagoras.....</i>	<i>12</i>
c.	<i>COMBAT XXI.....</i>	<i>12</i>
d.	<i>OneSAF</i>	<i>13</i>
B.	MILITARY GENRE IN GAME INDUSTRY.....	14
1.	Symbiosis Between Commercial Games and Military Simulations	14
2.	Importance of RTS Games.....	14
C.	ROLE OF OPEN-SOURCE SOFTWARE IN MILITARY SIMULATIONS	15
D.	SUMMARY	16
III.	SETUP OF THE RESEARCH CONFIGURATION.....	17
A.	ESSENCE OF THE COMPETITIVE APPROACH IN AI DESIGN	17
B.	THE DESIGN OF THE EXPERIMENT.....	18
1.	Using ORTS Game Engine to Create a Competitive Environment.....	18
2.	Creating a Scenario for the Competition.....	20
3.	Design of MOEs for the Experimental Competition	21
C.	CREATING AN INITIAL AI DESIGN.....	22
1.	Setting the Base Cases	22
2.	Creating a Competitive AI Design Based on Finite State Machine.....	24
3.	Creating a Competitive AI Design Based on Force Propulsion.....	25
D.	SUMMARY	27

IV.	IMPROVEMENT OF THE AI DESIGN BASED ON THE COMPETITIVE APPROACH.....	29
A.	RESULTS FROM THE EXPERIMENTAL COMPETITION.....	29
B.	MODIFICATIONS IN THE AI DESIGN	35
C.	RESULTS FROM THE SECOND RUN OF THE EXPERIMENTAL COMPETITION	38
V.	CONCLUSIONS AND RECOMMENDATIONS.....	43
A.	CONCLUSIONS	43
B.	RECOMMENDATIONS FOR FUTURE WORK.....	44
	LIST OF REFERENCES	47
	INITIAL DISTRIBUTION LIST	51

LIST OF FIGURES

Figure 1.	A battle between “UofA” and “Circle” designs.....	23
Figure 2.	FSM modeling a “marine” behavior	25
Figure 3.	Forces influencing a “marine’s” behavior	26
Figure 4.	The results from the battles between “Circle” and “g4a”	29
Figure 5.	The results from the battles between “Circle” and “g4b”	30
Figure 6.	The results from the battles between “UofA” and “g4a”	30
Figure 7.	The results from the battles between “UofA” and “g4b”	31
Figure 8.	Distribution of the time of battle in the battle between “UofA” and “g4a”	32
Figure 9.	Distribution of the time of battle in the battle between “UofA” and “g4b”	32
Figure 10.	Distribution of the casualties of “UofA” in the battle between “UofA” and “g4a”	33
Figure 11.	Distribution of the casualties of “UofA” in the battle between “UofA” and “g4b”	33
Figure 12.	A battle between “UofA” and “g4a” designs.....	34
Figure 13.	A battle between “UofA” and “g4b” designs	35
Figure 14.	Forces influencing a “marine’s” behavior for the “g4c” design	37
Figure 15.	The results from the battles between “Circle” and “g4c”	38
Figure 16.	The results from the battles between “UofA” and “g4c”	38
Figure 17.	A battle between “UofA” and “g4c” designs.....	39
Figure 18.	Distribution of the time of battle in the battle between “UofA” and “g4c”	40
Figure 19.	Distribution of “g4c” casualties in the battles between “UofA” and “g4c”	40
Figure 20.	Distribution of “UofA” casualties in the battles between “UofA” and “g4c”	41

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

2D	Two Dimensional
3D	Three Dimensional
AI	Artificial Intelligence
CAS	Complex Adaptive Systems
CBS	Corps Battle Simulation
COBRA	Combat Outcome Based on Rules for Attrition
FPS	First-Person Shooter
FSM	Finite State Machine
GPL	General Public License
ISAAC	Irreducible Semi-Autonomous Adaptive Combat
MANA	Map-Aware Non-uniform Automata
MOE	Measurement of Effectiveness
OSS	Open Source Software
RTS	Real-Time Strategy
SAF	Semi-Automated Forces

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

First and foremost, I would like to thank my family for everything they have done for me to pursue my dreams. I would like to thank Professor Christian Darken, my thesis advisor, for his guidance, patience, and countless advice. I would like to thank my classmates Brian Boone and Nabil Ouerghi, for their friendship, support and help all the time. I also would like to thank Commander Joseph Sullivan, Professor Rudolf Darken, Professor John Falby, all faculty and fellow students from the MOVES Institute, for their help and support.

To anyone I may have forgotten to mention by name, thank you!

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. MOTIVATION

In recent years, the focus of military operational analysis has been switching from massive conflicts, dominant in the Cold War, to local conflicts and local fighting, which are shaping the post 9/11 world. In the process of modeling this new type of warfare with Lanchester-type differential equations, the modeling community is facing some difficulties. These difficulties can be summarized by the following statements. Lanchester represents forces as homogeneous aggregate bodies, but to analyze a local battle or a force on force fight, the operation researchers need high-resolution models, with representation of each system or each soldier on the battlefield. In addition, these models must represent different tactics and behavior such as tactical movement and target acquisition and assessment.

Additionally, Lanchester's combat models do not represent command and control structures of the forces. Historically (Moffat, 2003), it has been proven that destroying the enemy's system of command and control significantly reduces his ability to fight and resist. The formulation of warfare as Lanchester-type differential equations does not include important components of the battlefield such as terrain or weather conditions. For example, the spatial layout of the forces has a significant impact on vital parameters of the battle such as visibility, ability to maneuver and range of fire. Lanchester's presentation of warfare does not include detailed human characteristics. In many cases, history proves that battles are won by morale, heroism, and courage, but not always by superiority in numbers.

In the search to address some of these constraints, combat modelers are trying to implement different solutions which in many cases employ solutions from the domain of artificial intelligence (AI). The main usage of AI solutions in military simulations is to model different tactics and behaviors of forces. This is done by using AI as a control

system for forces in the simulation, which are sometimes called synthetic forces. This allows the synthetic forces in the simulation to mimic the reactions and behavior of the real forces in the battle.

In most cases, the judgment of the success of the AI solutions is assessed by a comparison of how close an AI controlled entity's behavior is to a human controlled entity. This evaluation is normally made by subject matter experts who subjectively compare the expected entity behaviors to those shown in the military simulations. Although this approach is proven to lead to improvements, it does not provide enough information to answer questions such as: Why do humans prefer one or another strategy when they solve a particular problem in complex environments, especially in the military domain? Can the machines generate a better strategy for the same problem? Do additional factors exist that can be used to produce the AI solutions that can demonstrate more realistic or more natural behaviors.

In the search for alternative approaches, which may answer some of these questions, this thesis is focused on exploring how a competitive approach in AI design can be used to produce better AI solutions used in military simulations.

B. RESEARCH QUESTIONS

The research questions that support the thesis statement are:

1. Are real-time strategy (RTS) games a useful framework for military simulation AI design?
2. Does the ORTS game engine, an open source RTS framework, provide an adequate environment for testing a competitive approach in AI designs?
3. Can a competitive approach be used for better understanding of a combat situation and eventually lead to better AI solutions used in military simulations?

C. SCOPE

The scope of this thesis is limited to exploring a military scenario in a RTS game where different AI designs are competing against each other. By analyzing the results of this competition, this thesis will attempt to identify directions for the improvement in AI design.

D. THESIS ORGANIZATION

This thesis is organized into the following chapters:

Chapter I: Introduction. This chapter provides the motivation, the research questions, and the framework of the thesis.

Chapter II: Background. This chapter is a literature review of the research conducted in the field of military simulations, AI, and the game industry.

Chapter III: Setup of the research configuration. This chapter describes the experimental setup for testing the competitive approach in AI design. It shows how the ORTS game engine and an ORTS game scenario can be used. It explains the development process of the initial AI design with corresponding limitations, considerations, and decisions.

Chapter IV: Improvement of the initial AI design based on the competitive approach. This chapter contains an analysis of the results from the experimental competition, possible directions for improvement of the initial AI design, and results from the second experimental competition.

Chapter V: Conclusions and recommendations. This chapter wraps up the results from the thesis research. It contains general conclusions and recommendations for future work.

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND

A. APPLICATION OF AI DESIGNS IN MILITARY SIMULATIONS

1. Traditional Approaches in Military Simulations

a. Lanchester's Formulation of Combat

Formalization of the rules of warfare has been a subject for research for a long time. In 1914, Lanchester was the first researcher to develop the description of modern combat as a set of differential equations (Taylor, 1983). In his original paper, Lanchester distinguished two types of combat: ancient combat, which was described by Lanchester's linear law and modern combat, which was described by Lanchester's square law. In the model of ancient combat, units are engaged in one-on-one duels. This results in the casualty-exchange ratio being independent from the number of units on the opposing side. In modern combat, it is possible to have situations in which many units are engaging a single unit. This is the case of an aimed fire, where individual targets are identified and attacked by any number of opponent units. The model of modern combat explains the significance of tactical advantages, such as force concentration and supports Julius Caesar's famous dictum "divide and conquer."

To fully analyze Lanchester's combat models, some additional assumptions must be considered. Lanchester's combat models are purely hypothetical. Although they have resemblance to real battles such as ancient duels or aimed fire, they do not represent any particular battles or historical events. For simplicity, the model assumes the battle is fight-to-the-finish until one side is totally annihilated at the end, which rarely ever has happened in any historical battle. Lanchester assumes homogeneous forces. When considering the fighting forces, both sides are represented by unified units with the same capabilities to fight and to sustain. The casualty-exchange

ratio or attrition coefficients are assumed to be constant. Although attrition coefficients for ancient combat and for modern combat are different, they do not change during the battle.

The reasons why Lanchester's combat models are broadly used in military simulations and are considered the classic representation of warfare can be summarized by the following points (Bracken et al., 1995; Taylor, 1983). Lanchester suggested a very simple model. First, to create a new combat model it is only necessary to know the attrition coefficients, and the initial number of forces. Next, the model's differential equations can be easily solved in functional form. Lanchester's combat models can be validated and results can be compared to historical data. For example, in his research, Lanchester had discussed the battle of Trafalgar in 1805 and the tactics of Admiral Lord Nelson. Lanchester's model accurately reflected the outcome of the historic battle. Other authors, such as Lucas and Turkes (2003), also found historic evidence for the validity of Lanchester's models.

The suggested combat models are flexible enough that new tactics can be easily incorporated. For example, some of the extensions of Lanchester's combat models are formulation of mixed combat (ambush) or heterogeneous combat. The biggest achievement in Lanchester's formulation of combat is the translation of the verbal model into mathematical terms. In this manner, he was able to justify the principle of force concentration (Taylor, 1983). Solving Lanchester's differential equations gives important information of the dynamics of the battle and answers questions such as:

- Who wins?
- How many are casualties and how many survive on each side?
- What is the time duration of the battle and what is the size of remaining forces in any given time?
- What are desired initial number of forces and attrition coefficients, so the victory can be guaranteed?

b. Enrichments of Lanchester's Models

In attempts to address some of the limitations of Lanchester's combat representation, researchers introduced some enrichment to the original models. For example, modifications were made to represent the different ending conditions or introduction of reinforcements (Bracken et al., 1995; Taylor, 1983). Another enhancement, especially in the area of naval warfare, is Hughes' formulation (1995) of salvo equations (Bracken et al., 1995).

The next major step of combat modeling development was the stochastic Lanchester-type presentation (Taylor, 1983). The stochastic models are based on the fact that combat is not a deterministic process. Numerous examples that illustrate this non-deterministic nature of warfare can be found in history. Even Lanchester based his models on the assumption of "on average," which implies that the foundation of combat modeling is stochastic processes.

The idea behind stochastic models is to divide the time of the battle into very small intervals and then analyze the possible outcome in each of these intervals using probability theory. In the model, combat is presented as a continuous time Markov Chain, whose states are the number of survivors on each side. Based on this assumption, the original Lanchester's model can be interpreted as being derived from the Markov model simply by replacing all random variables by their expected values. The main advantage of this approach is that it addresses some sources of uncertainty in a battle such as a time and a place of a casualty occurrence. In addition, it can be easily presented as a time-step Monte Carlo simulation.

The biggest problem of a stochastic combat presentation is the exponential increase of model complexity. Even though computers are used in model calculation, the results are approximate and it takes time to be produced. Another consideration is that differential equations need to be solved only once, but the stochastic simulation must be run many times to produce the data, which is necessary to analyze the variability of combat results.

Another approach to provide more realistic and adequate combat models is to introduce rule-based modifications in the Lanchester-type equations. For example, in Corps Battle Simulation (CBS) a tool called Combat Outcome Based on Rules for Attrition (COBRA) was created (Henry, 1994). The idea of COBRA is to use a rule-based expert system to analyze the combat situation and the preexisting force-on-force attrition algorithms, which are updated with integrated combat factors such as: mission, enemy, terrain, troops or time planning.

In conclusion, despite the fact that the Lanchester's original goal was to develop a mathematical argument that supports the general tactical principle of concentration, actually he developed a methodology that gives reasonable quantitative results for combat modeling. The best evidence for the validity of his theory is the fact that it is still in use for operations research and combat modeling over 70 years after its inception.

2. AI Based Approach in Military Simulations

a. Necessity of New Paradigm in Combat Modeling

By analyzing current combat models and historical data, researchers found situations that cannot be explained by Lanchester's models. For example, Lauren (1999) emphasized "nonlinearity" in modern warfare. He argued that Lanchester's models do not explain cases in which a "superior" force is defeated by an inferior force. Lauren considered that nonlinearity is caused by factors on the battlefield that are not represented in Lanchester's combat models. As an example, he shows how maneuver and morale can dramatically affect the outcome of a battle.

Another consideration in Lauren's critique is that in many cases the attrition coefficients are adjusted to fit to historical data, and in general human behavioral factors are ignored. Furthermore, he argues that this approach does not guarantee that historical data can be used to predict the likely outcome of battles, given the change in tactics and technology.

The problem is even more complex when one models asymmetric warfare or operations other than war. Usually in those operations, the goal of the operation is to accomplish the mission with the minimum number of casualties on either side. In addition, in operations where the mission is reconstruction, there are no casualties expected at all. Different kinds of measures indicate the success or failure of the operation. This type of mission totally contradicts Lanchester's idea of the attrition coefficients.

The solution suggested by Lauren is to change the paradigm of combat modeling and to be more specific, by exploring the complexities of warfare and treating it as a complex adaptive system (CAS). Mainly, his idea is to use a bottom-up approach, where the individual combatants are modeled, and their interaction in a battlefield will produce desired data for combat analysis. This idea is very close to the idea presented in the research of Ilachinski (2000) on Irreducible Semi-Autonomous Adaptive Combat (ISAAC)/EINStein models.

b. Ilachinski's Combat Models

Ilachinski found a couple of similarities between combat and complex systems. Some of the similarities are nonlinearity, hierarchical structures, decentralized control, collective dynamic, self-organization, adaptation and emerging behavior. His next step was to explore combat as a complex system and to create the ISAAC simulation.

In essence, ISAAC is a simulation where abstract discrete homogeneous agents (combatants) interact with each other, following the rules of cellular automata. In this virtual combat, each agent has its own set of characteristics and rules of behavior, which can evolve over time. Examples of these characteristics are state (alive, injured, or killed), mission and ability to adapt. Another important element in ISAAC is the notion of the agent's personality. By incorporating agent personalities, Ilachinski created the driving force for each agent. This is the mechanism defining the agent's decision to fight, to retreat, or to help a team member.

Running ISAAC simulations, even with this simple model, Ilachinski managed to achieve emerging behaviors, which can be classified similarly to frontal attack, local clustering, retreat, penetration, flanking maneuvers and more. Despite the fact of these encouraging results, Ilachinski made a clear statement that this behavior should not be treated as exact behavior. It is similar to real behavior, but it does not have all the complexity shown on the battlefield. As such, words such as “retreat” or “flanking” should be used just as a reference.

The biggest achievement of Ilachinski’s research is that he managed to switch the main question of combat analysis from “Who won?” to “What if?” by opening a completely new dimension – agent-based modeling. Despite the fact that this is a relatively new area of research, the results are quite promising.

c. Advantages of AI in Agent-Based Combat Models

Models based on principles of cellular automata show good results when they are used to represent behaviors shown in the natural world or in social life (Miller & Page, 2007). A demonstration of this success is Reynolds’ work (1987) on animating a flock of birds. Further exploration of Ilachinski’s models in advanced combat modeling certainly is facing some limitations. An example for these limitations is modeling the behavior of the modern soldier. Today, he is equipped with advanced combat systems, in a position of constantly making critical decisions while flooded with information. In this situation, simplification of this soldier as a simple reactive agent can skew the results and even can completely compromise the combat model.

One way to eliminate some of the limitations of cellular automata is to enhance the agent models by introducing some AI elements. In the current literature (Russell & Norvig, 2003) the subject of AI is defined as the exploration of systems that act or think as humans or act and think rationally. The AI researchers are interested in topics such as natural language processing, knowledge representation, automated reasoning or computer vision. All these topics are part of the cognitive processing experienced by soldiers on the battlefield. The question then becomes how to create better models representing the cognitive process of soldiers in the battlefield. To address

this question, AI research is focused on the interior of rational agents. The interior is the key factor in defining the complexity of tasks that an agent can accomplish. For example, reactive agents are making their decisions only on current perception, and everything in the environment that is not perceived can cause serious trouble for the agents. Learning agents do not have this problem, but this type of design requires significant computational power to be able to execute complex cognitive algorithms.

In conclusion, it can be stated that using AI in combat modeling is still an open question, and a lot of research is focused on finding an appropriate AI design for the different types of combat elements. Some good examples of AI solutions used in combat modeling are ongoing research on tactical level combat and tactical soldier movement (Back, 2002, Cox & Fu, 2005, Mueller 2006, Pawloski, 2001, Reece et al., 2000).

3. Military Simulations Implementing AI Solutions

a. MANA

The Map-Aware Non-uniform Automata (MANA) software is developed by New Zealand's Defence Technology Agency (Lauren & Stephen, 2002). MANA is considered to be a continuation of the work on Ilachinski's ISAAC/EINStein model. In MANA, Ilachinski's ideas are used, but some improvements have been made. For example, the entities communicate not only with their neighbors, but also on the squad or on the formation level to achieve the desired goal. MANA has an extensive set of prebuilt entities, which allows rich combat scenarios to be created. The set includes entities such as individual soldiers, vehicles or fixed wing aircraft. In addition, each entity has a broad list of properties, which make it adjustable for different requirements. Entity movement is done with the help of objective waypoints. The terrain is presented as a bitmap, where a different coloring is used to represent the features on the map, such as roads, obstacles or areas that cannot be passed.

The goal of MANA is not to provide "magic" number answers, but rather to provide insight on the dynamics of the battle and its most significant factors. MANA has proven to be a valuable tool in modeling asymmetric warfare and logistics operations.

b. Pythagoras

Pythagoras is an agent-based simulation originally developed by Northrop Grumman to support a U.S. Marine Corps-sponsored international initiative focusing on human factors in military combat and noncombat situations (Bitinas et al., 2003).

Pythagoras is platform-independent and is written in Java. It can be run as a batch job on a cluster of computers, enabling thousands of replications to be executed in a short time. Pythagoras is unique in the sense that it introduces fuzzy-logic and soft decision rules in agent decision-making. This is an attempt to turn commander's orders, in a written format, into numeric format.

Some of the advantages of Pythagoras are simple scenario development and flexibility to incorporate abstract ideas. At the same time, it is difficult to validate the models made in Pythagoras, which is a common problem found in all agent-based simulations.

c. COMBAT XXI

COMBAT XXI is a joint Army-Marine Corps effort to replace the Combined Arms and Support Evaluation Model (CASTFOREM), a legacy combat simulation (Posadas, 2001). It is a non-interactive, high-resolution, analytical combat simulation focused on the tactical level. Combat XXI is being developed by the U.S. Army TRADOC Analysis Center -White Sands Missile Range (TRAC-WSMR) and the Marine Corps Combat Development Command (MCCDC). COMBAT XXI is written in Java and uses SimKit as a simulation engine.

Each of the simulated entities consists of a decision component and a functional component. The decision component is a set of decision interaction modules, which define the response capabilities of each entity to each simulation event. The functional component creates an interface to physical algorithms. The set of functional modules defines how a simulated entity will perceive and interact with the outside world. In addition, COMBAT XXI provides a global mediator mechanism, which creates the

next level of abstraction in interaction mechanism between entities. Examples of these mediators are the observe mediator and the damage mediator.

Additional important concepts in COMBAT XXI are pyramidal structure behavior models (primitive, procedural, tactical, and cognitive), the introduction of stochastic elements in the Observe-Orient-Decide-Act (OODA) decision cycle, and the presence of a scripting capability for extending behaviors. All these make COMBAT XXI a much more realistic and advanced combat modeling tool. COMBAT XXI is expected to be a universal analytic combat modeling tool for the U.S. Army and U.S. Marine Corps.

d. OneSAF

The design goal of the OneSAF Objective System is to provide a Semi-Automated Forces (SAF) architecture that incorporates and expands existing SAFs (Henderson & Rodriguez, 2002). OneSAF will be a system that allows modeling and simulation of a full range of military operations, systems, and control processes. In addition, OneSAF will support models of various fidelity and resolution levels, suitable for different uses, including both training and analytical objectives, and at the same time it will be able to exchange information with other simulations.

The key for building OneSAF is the component based approach, providing separate services for presentation, composition and execution. Special attention is focused on behavior models. Entities in OneSAF are classified as physical agents, behavior agents and units, where each can have a set of primitive and composite behaviors. In addition, OneSAF provides timelines, rule sets and behavior metadata, which allow the agents in OneSAF to exhibit qualities such as autonomy, collaboration, adaptability and mobility. OneSAF is expected to play a major role in the future combat modeling for training purposes.

B. MILITARY GENRE IN GAME INDUSTRY

1. Symbiosis Between Commercial Games and Military Simulations

Lenoir and Lowood (2002) state that for a long time, commercial games have shaped the design of military simulations. The evidence of symbiosis between commercial games and military simulations can be traced back to the early 1880s in a form of wargames for officer training. History has proven the idea that a game can be a valuable tool of military education, and in many cases, only a game can give military professionals the sense of the dynamics and complexity of combat.

The development of computer technology gave a new impulse to the marriage between commercial games and military simulations. A good example is the project for distributed military simulation network, SIMNET. The design goal of SIMNET was to create a network of simultaneously running military simulations focusing on meeting the training objectives, rather than trying to achieve the highest possible physical fidelity. This idea was captured and enriched by the game industry in very popular online multiplayer games such as “World of Warcraft,” “Dark Age of Camelot,” Ultima Online” or “Guild Wars.” At the same time, the transfer of ideas and technologies is not unidirectional. For example, “America’s Army,” a first-person shooter (FPS) game used for training in the U.S. Army employs a game engine originally designed for a commercial game “Unreal Tournament.” In the future, the gaming format it expected to be more broadly used in military education, blurring the line between simulated and real combat.

2. Importance of RTS Games

Adams (2006) defines an RTS game as "a military strategy game in which the primary mode of play is in a real-time setting." Therefore, a RTS game can be treated as an abstraction of a conflict between two or more sides, each of which is trying to achieve total dominance through numerous battles of virtual forces.

Using this definition and other analysis on RTS games (Adams, 2006; Devkar, 2003; Keefer), some similarities between RTS games and military simulations should be

considered. Both RTS games and military simulations use the same principles for combat modeling. Although the military equipment is not presented in RTS games with real parameters, they sometimes try to be as realistic as possible. Both RTS games and military simulations have distinguished sides where players play against each other, but there are no formal constraints in making alliances or coalitions. The terrain is a key element for both RTS games and military simulations, but in RTS games it usually is not referenced to any real place. For both RTS games and military simulations, graphics is an important element, but in many cases it is the graphics that define the financial success of a RTS game, not the analytical results.

At the same time, RTS games have unique qualities that are important for AI developers. The RTS game scenarios are very complex. In addition to combat, they can include activities such as resource management, construction or trade, which makes RTS games an excellent environment for exploring AI solutions for a variety of tasks. RTS games are very popular entertainment, which has created a big community of users who are evaluating AI designs every day by buying or playing RTS games. RTS games are not constrained to represent physical objects with exact parameters, such as weapons or vehicles, so the designers have new levels of abstraction which are focused only on AI algorithms.

Another important feature of RTS games is the notion of real-time. Usually, this means the players in RTS games do not have to wait for an opponent's move. This creates situations close to real warfare, where the commanders must assess and respond to changes on the battlefield as quickly as possible. Real battles prove that, in many cases, a decision in the right time is more important than the best decision at the wrong time. This real-time feature of RTS games is a serious challenge for the AI designs. AI must be able not only to accomplish the desired goal, but also to do it as quickly as possible.

C. ROLE OF OPEN-SOURCE SOFTWARE IN MILITARY SIMULATIONS

The study conducted by MITRE Corp. (2003) shows the importance of using Open-Source Software (OSS) in the U.S. Department of Defense. In the study, there is a

clear distinction between freeware and OSS. Mainly freeware is focused on the cost and can be qualified as zero-cost software, whereas the emphasis of OSS is on the ability for users to study, change and improve the source code.

OSS is attractive in military simulations for a couple of reasons. OSS creates a new platform for the exchange of ideas. Free access to source code allows more people to be involved, to collaborate and to produce better software. Implementing OSS in military simulations significantly reduces development costs. In addition to the fact that usually OSS is free of charge, it increases competition and forces industry to reduce the cost of proprietary software.

The most valuable aspect of OSS for military simulations is the establishment of open standards. It is proven by practice that the future belongs to integrated solutions, where well structured modules work together to achieve the system's goal, and open standards are the only way to produce and to manage this kind of simulations (Buss, 2002).

Some good examples of OSS being used in military simulations are SimKit (Buss, 2002) and Delta 3D (Darken et al, 2005). At the same time, OSS should not be considered as the universal solution. In many cases, OSS products are not market oriented and the lack of good documentation or support discourages end-users.

D. SUMMARY

This brief overview of the research in combat modeling confirms the importance of exploring new methods for improvement of AI designs used in military simulations. The question is more significant because the results and techniques used in the modeling of modern warfare can be applied in many other domains with complex and non-linear properties.

III. SETUP OF THE RESEARCH CONFIGURATION

A. ESSENCE OF THE COMPETITIVE APPROACH IN AI DESIGN

The idea behind the competitive approach is to treat the development of new AI designs for military simulations as an optimization in an abstract hyperspace. This optimization is an interactive process, with the following steps:

- Create or choose an AI solution for the base case.
- Create or choose another AI solution for comparison with the base case.
- Compare these two AI solutions by putting them in a competitive environment.
- Analyze the results from the competition and use them to create a new AI solution for the next iteration.

To be successful, this optimization procedure should employ the following conditions:

- The adequate environment and scenario must be set, allowing the AI designs to demonstrate their capabilities.
- A set of measurements of effectiveness (MOE) must be created, which will be used for quantitative evaluation of each AI design.
- In each step in the optimization, only feasible AI solutions must be produced. This means that each newly generated AI design must be able to achieve its objective and should be free of errors.

There must be a clear distinction between a battle simulation run in a military simulation and a battle simulation used in competitive approach. Although they both represent force-on-force conflicts, the configuration of forces are significantly different. In the military simulations, forces have the same AI driving designs and the winning side is determined by differences in additional parameters such as ability to move or weapons fire range. In battle simulations using the competitive approach, the parameters of forces are the same, and the driving AI design makes the difference in the battle. In this way, the competitive approach allows for the designing of experiments, in which the AI design is the independent variable, and the response is to be measured by selecting MOEs.

B. THE DESIGN OF THE EXPERIMENT

1. Using ORTS Game Engine to Create a Competitive Environment

The first step of the research was to find an appropriate environment for the testing of the competitive approach in AI designs.

The true military simulation platforms such as MANA, Combat XXI or OneSAF were not used because of the following considerations:

- Current military simulations are very complex environments and testing a new idea or new approach can cause unwanted simulation crashes instead of showing the desired results.
- In many cases, military simulations are not open source projects and making changes in the entity tactics or behavior is not easy.
- To prove the conception of the competitive approach in AI design a simple type of warfare is needed, and running fully loaded military simulations will take more resources and time than necessary.

By analyzing and comparing products that are available on the market and products similar to military simulations, the Open Real-Time Strategy (ORTS) game engine project attracted this author's attention. ORTS has been developed by the University of Alberta, Canada (Buro, 2002; Buro, 2003; Buro, 2004; Buro & Furtak, 2003; Buro & Furtak, 2004; Buro & Furtak, 2005; Buro et al, 2005).

The ORTS game engine has a client-server architecture. The server is responsible for managing the state of the "world" of the game, and the clients are responsible for analyzing the current situation in the game and responding adequately by sending commands back to the server. To enforce fairness in the game, the ORTS server executes received commands in random order. In the OTRS game, each client can control only the assets that it possesses, and depending of the scenario of the game, may or may not have full information about the opponent's assets. The number of clients is equal to the number of opponents in the game. The properties of terrain and the assets vary, depending on the game scenario.

Some of the reasons for choosing ORTS as a research platform are:

- ORTS was built mainly as a research project for studying real-time AI problems such as pathfinding, scheduling, planning and dealing in situations with imperfect information.
- ORTS is free software distributed under General Public License (GPL).
- ORTS is written in C++ and all the code and protocol descriptions are freely available. This allows full freedom to implement different AI solutions.
- The client-server architecture of ORTS allows a single non-controversial state of the world for the game at a single time with almost unlimited player sides, human or AI driven.
- The ability to choose from different graphic clients, such as 2D or 3D, can significantly reduce polygon rendering without losing important information.

Some additional advantages of ORTS as a competitive environment are:

- ORTS has a built-in tournament manager, which significantly reduces time for the design of experiments and analysis of results.
- ORTS has pre-built game scenarios, which are rich enough for exploring AI designs demonstrating complex behavior.
- The designers of ORTS encourage development of AI solutions for tournament participation in a broad community, including industry, academics, and just hobbyists.

At the same time, ORTS has some disadvantages, which can be summarized as the following:

- Not all ORTS games have military relevance or military applications.
- ORTS started as a student's experimental project and there are parts of the code which are not fully debugged and documented. Also, in some cases the functionality which is implemented is not stable, which leads to crashes.
- As with many open source projects, ORTS lacks the proper documentation, which makes the learning curve for new programmers very steep.
- Although the designers claim that ORTS is platform independent, porting the code to another operating system without losing functionality is almost impossible. For example, ORTS versions for Windows (Visual Studio, CYGWIN, MinGW) run very slowly if they even compile.

A central concern for ORTS as it relates to this research is that it cannot be treated as a deterministic simulation. ORTS is based on the network distributed architecture, and this introduces an additional randomness in the ORTS event execution mechanism. This randomness is caused by two factors: the nature of TCP/IP protocol and the operating system.

First, TCP/IP protocol uses random mechanisms to acquire the transmission medium, so this randomness is inherited by ORTS. Second, the operating system is a source of randomness, if the server or the clients in an ORTS game are running on the same machine. Usually, the operating system treats them as different processes, and because ORTS does not provide exclusive mechanism for resolving concurrency, the threads are executed randomly.

Although this limitation can be minimized in statistical analysis, the problem with manipulation of the seed in a simulation remains. In an ORTS simulation, the same simulation seed does not produce the same result. By setting the simulation seed, identical conditions can be reproduced for the terrain or for the initial layout of resources, but because of its inherited randomness, ORTS will produce different results each time. Therefore, the manipulation of the simulation seed cannot be used as a replay mechanism for behavior analysis in ORTS. All these drawbacks are relatively easy to overcome, and they have no significant impact on the results of the thesis research.

2. Creating a Scenario for the Competition

From all built-in games in ORTS, the game number four — “Small-Scale Combat” — was selected as the best representation of generic combat. For this game, the setup of the scenario is as follows:

- There are two opponent sides.
- Each side starts with fifty randomly positioned soldiers (“marines”).
- The terrain is without obstacles, spread over sixty-four by forty-eight tiles. The “marines” can be positioned on a finer grain than a single tile, which means that a couple of “marines” can occupy a single tile.
- Each side has perfect information for position and status of its opponent’s soldiers.

- Some small mobile obstacles ("sheep") are moving randomly. They cannot be moved or destroyed by the soldiers.

The objective of the game is to destroy as many opponent "marines" as possible within a five-minute time limit.

In the ORTS server, the "marines" are presented as objects with the following characteristics:

- In the beginning of the game, each "marine" has eighty life points.
- When it is hit, the "marine's" life points are decreased by a random number between five and seven.
- The "marines" are free to move in each direction, but they are limited in maximum speed.
- The "marines" can move and attack simultaneously.
- The "marines" do not have to face the target before shooting.
- The "marine's" weapon has a certain firing range, and it needs to cool-down before its next fire.

The goal for the AI design is to create an ORTS client that shows complex behavior and implements winning tactics. Challenges for the AI design in this scenario are the tactics used in small-scale combat, cooperation in goal achievement and unit management. In addition, all other challenges that a RTS game can offer for AI research are present, including the real-time phenomenon.

3. Design of MOEs for the Experimental Competition

The main MOE, which measures the success or failure of each AI design, is the percentage of wins. It can be calculated using the following formula:

$$\% \text{ wins} = \frac{\text{number_of_wins}}{\text{number_of_games_played}} \quad (1)$$

Although this MOE has great expressive power, it is not sensitive enough. For example, if one of the AI designs is sufficiently dominant, it will start winning all the time. The results will show its dominance, but there will be no information why this is happening. Therefore, two new, more sensitive MOEs are added: the number of

casualties at the end of the battle and the duration of the battle. This set of MOEs allows each AI design to be positioned according its performance, and at the same time gives enough information about possible advantages or weaknesses of each AI design.

To collect the data for these MOEs, the Tournament Manager (TM) was used. This is a built-in feature in ORTS, which allows a tournament of battles to be conducted, where the AI designs fight with each other in a full factorial combination. Each battle can be repeated an unlimited number of times with different simulation seeds. The ORTS TM is capable of recording the time when a battle ends and the casualties for both sides. The output from the ORTS TM is a text file, which makes the reported data available for processing with almost all statistical packages.

C. CREATING AN INITIAL AI DESIGN

1. Setting the Base Cases

The design of the first version of an ORTS AI client was based on results from the ORTS 2007 tournament (ORTS, 2007). At the time when the thesis research had started, only two AI designs for the ORTS game four were available. These were the NPS entrant in the tournament, developed by Patrick Jungkuz, and the design developed by the University of Alberta. For convenience, each tested AI design was assigned a codename (and is referenced by its codename later in the thesis). The codename for the AI design for the NPS entrance in the ORTS 2007 tournament is “Circle,” and the codename for the AI design developed by University of Alberta is “UofA.”

Having these two AI designs, the research was focused not on analysis of the source code, but to treat these AI designs as black boxes and analyze their behavior in terms of strengths and weaknesses.

“Circle” demonstrates a tactic based on Lanchester’s rule for “force concentration.” In the “Circle” design, the “marines” create a formation in a set of concentrated circles, and that is why the design is called “Circle.” A “Circle” formation is shown on the right in Figure 1, presenting a snapshot of a battle between “UofA” and “Circle” designs.

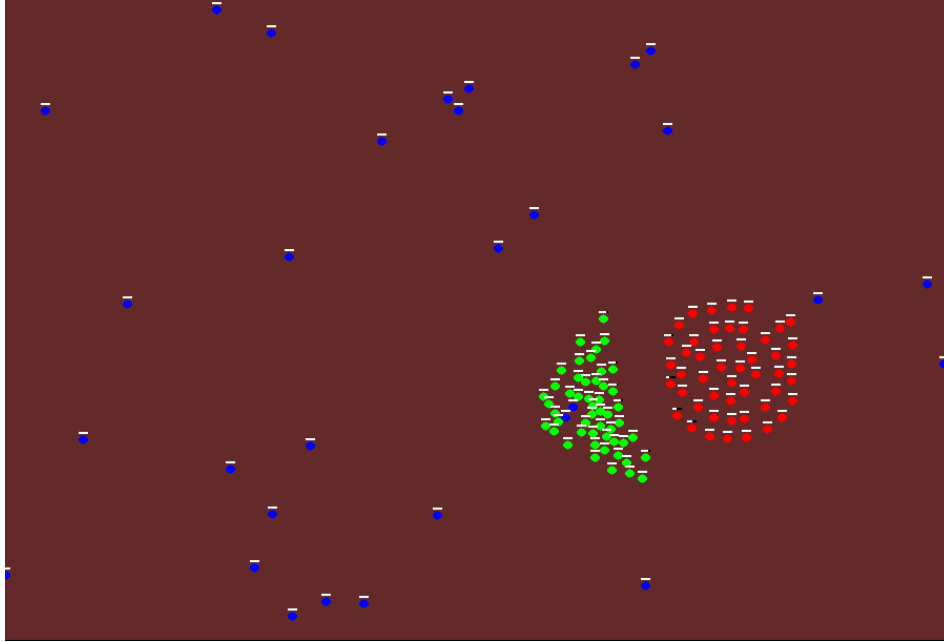


Figure 1. A battle between “UofA” and “Circle” designs

During the battle, the “marines” try to stay in a circle formation, and at the same time, constantly try to allow new (“fresh”) forces to fire as much as possible. This is done by rotating the whole formation. If the enemy is destroyed in the perimeter defined by the weapons range, the “marines” try to pursue the opponent, again within the circle formation.

The ORTS 2007 tournament showed that the design of “UofA” was more advanced than the “Circle” design. “Marines” controlled by “UofA” also use Lanchester’s rule for “force concentration,” but the formation is more flexible. “UofA” moves “marines” with an edge formation facing the center of the opponent’s forces. A “UofA” formation is shown on the left in Figure 1. During the battle, “UofA” tries to outflank the opponent’s forces, limiting their ability to maneuver. This positions more “UofA” soldiers within firing range, and usually this is the key for winning (again reminiscent of Lanchester’s model). The results from ORTS 2007 tournament (ORTS, 2007) showed that “UofA” does not have the best AI design, but demonstrates total dominance over “Circle.”

2. Creating a Competitive AI Design Based on Finite State Machine

The next step in the competitive approach is to create a competitive AI design that will hopefully be, or become, better than the base cases. The challenge for the chosen scenario is in two categories. The first part is movement. The soldiers must be in the right place, and in the right time, when they fire on the opposing forces. Therefore, the AI design must provide an advanced tactical movement. The second part is another optimization problem. The problem is to decide when the soldiers are in a firing position, and with which opponent to engage, so the salvo will have maximum effect. In its essence, the problem is a fire allocation problem, and solving it is not a trivial task, so it is out of the scope of this thesis. In this thesis, heuristic algorithms are used to approach this problem, with an emphasis on quick results.

The research in the thesis was focused on improving soldier's tactics through improvements in movement only. To achieve "marine" tactical movement, the test design was separated into two pieces with codenames "g4a" and "g4b." The goal was to create two AI designs with different architectures, and by using the competitive approach to distinguish which one is capable of better accomplishing the task in the ORTS game four.

The idea of AI design for "g4a" was to create more "advanced" soldiers, by using the principles in CAS and Ilachinski's work, which will create new emergent behavior, and this collective behavior will bring the victory. Therefore, the attention was focused on the interior of the individual "marine."

One way to model soldier's behavior is to use Finite State Machine (FSM); the design of such an FSM modeling a "marine" behavior for "g4a" is shown in Figure 2.

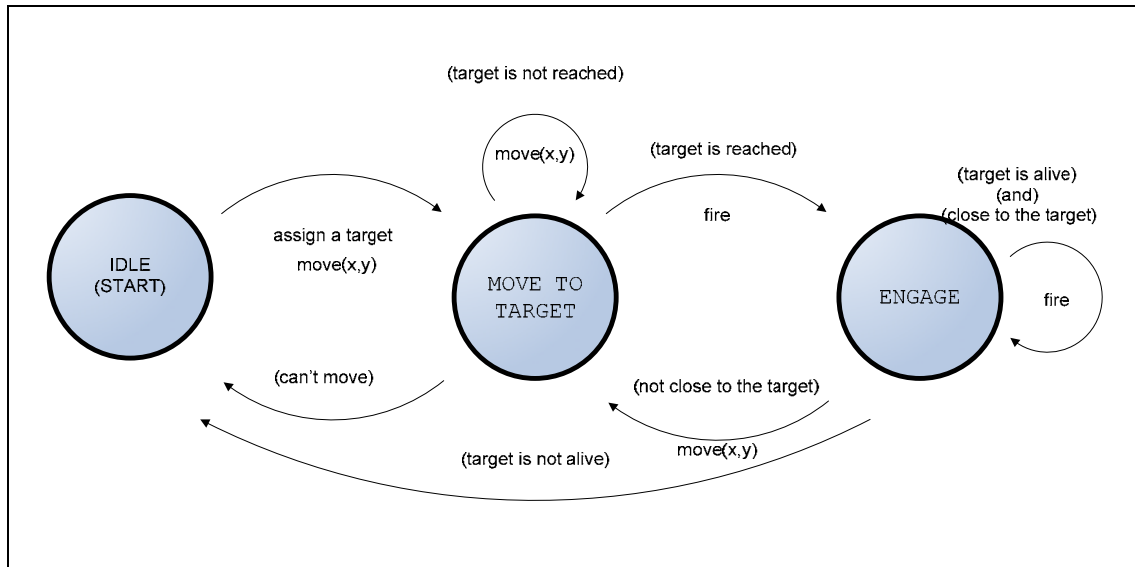


Figure 2. FSM modeling a “marine” behavior

Despite the fact that it has only a few states, it is powerful enough that each “marine” can fight and escape obstacles and accomplish his mission.

The overall coordination of the “marine” force is achieved by the information provided to individual “marines.” Because the ORTS game four is a game of perfect information, each “marine” can make his own decision on which opponent to engage, and when to engage him. Another advantage of using FSM is that it allows the “marines” easily to escape “sheep” or a friendly “marine” standing in their way. The AI design for the “marine” can be classified as model-based reflex agent (Russell & Norvig, 2003).

3. Creating a Competitive AI Design Based on Force Propulsion

The goal for AI design used in “g4b” was to test an alternative to the approach used in “g4a.” As was mentioned, “g4a” is based on a bottom-up approach, where the goal is achieved without central coordination. Contrary to the bottom-up approach of “g4a,” an artificial component, a “commander,” was introduced in the design for “g4b.” The purpose of this “commander” is to get the picture of the battlefield and to send commands to each “marine.” At the same time, this was implemented indirectly. “Marines” are still individual agents, but it is the “commander” who controls the driving

force for each agent. This can be classified as more of a hybrid approach than the “g4a” design. The principles of CAS are still in place, but in a more controlled and predictable way.

To achieve “force concentration,” the design of “g4b” employs the following principles, which are used in the “flocking” algorithms developed by Reynolds (1987). To get simulated birds in a nice “flocking” formation, Reynolds uses pure-reflex agents (Russell & Norvig, 2003), where the sum of impact forces creates the vector of driving force. In “g4b,” each “marine” is also a pure-reflex agent but because the “marine” formation is not a flock, “marines” have a different set of impact forces. This set is shown in Figure 3. The top two arrows are representative of the attractive forces, and the bottom two arrows are representative of the repulsing forces.

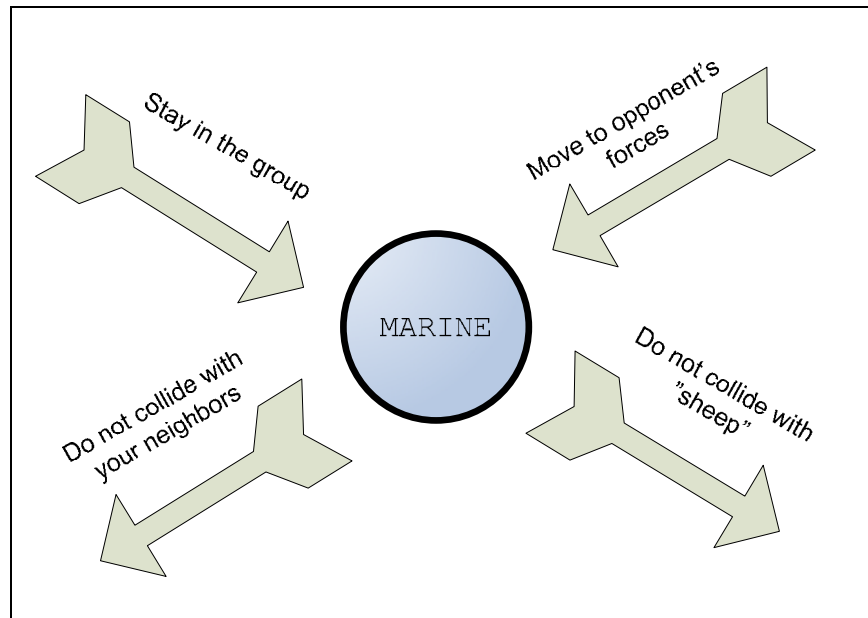


Figure 3. Forces influencing a “marine’s” behavior

This small set of four forces allows the marines to stay in formation and at the same time to approach the opponent’s forces. As opposed to the “Circle” formation, the “marines” in “g4b” are in a dynamic and more flexible formation. In this way, they do not have to wait for formation to be built, but rather are in formation all the time. This formation is constantly changing in shape because there is no condition to stabilize it.

D. SUMMARY

The configuration of the experimental design allows objective measurement of performance by putting AI designs in a competitive situation. The use of the ORTS game engine creates an environment flexible enough that different AI designs can be tested and evaluated without major pre-configuration. Furthermore, ORTS allows any additional information to be collected automatically, if it is needed for future analysis.

The nature of the two initial AI designs is distinct enough that the dominant factors contributing to the winning strategy can be detected, and eventually enough data will be collected that improvements in AI design could be possible.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. IMPROVEMENT OF THE AI DESIGN BASED ON THE COMPETITIVE APPROACH

A. RESULTS FROM THE EXPERIMENTAL COMPETITION

The first set of experiments ran a set of battles between the base case, “Circle,” and test cases, “g4a” and “g4b.” The ORTS TM was set up to run each battle a hundred times with a different simulation seed each time. The corresponding results for the main MOE, the percentage of wins, from the battles between “Circle” and “g4a” are presented in Figure 4, and the results from the battles between “Circle” and “g4b” are presented in Figure 5.

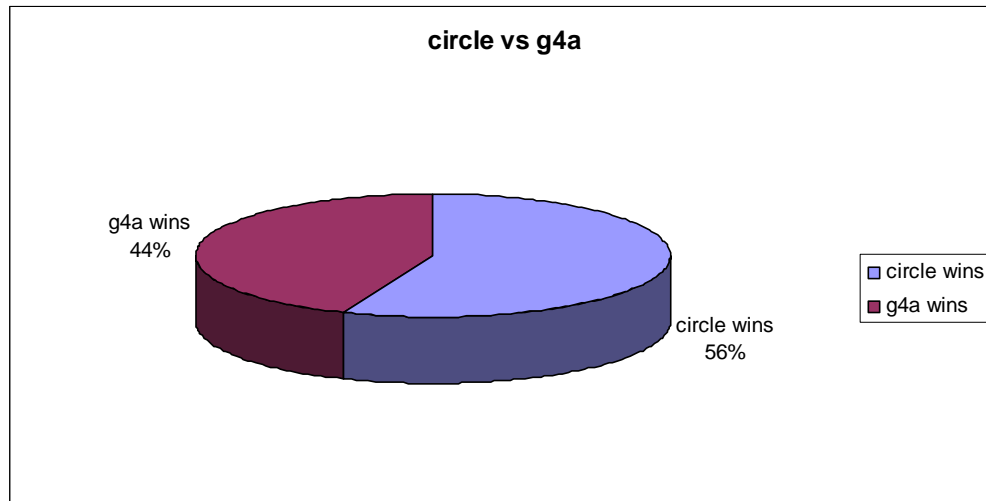


Figure 4. The results from the battles between “Circle” and “g4a”

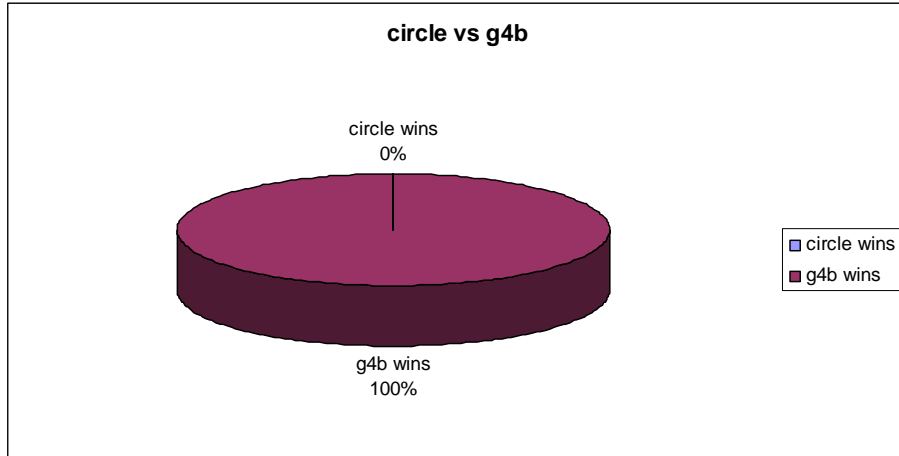


Figure 5. The results from the battles between “Circle” and “g4b”

By analyzing the chart in Figure 4, it is obvious that the design of “g4a” does not fulfill the desired goal. It obtained only forty-four percent of the wins, and this is clear evidence that “Circle” had better performance than “g4a.” On the other hand, Figure 5 shows that “g4b” had total dominance over “Circle.”

The same experiment was executed again, but this time, the battle was between the base case, “UofA” and test cases, “g4a” and “g4b.” The results for the main MOE, from the battles between “UofA” and “g4a” are presented in Figure 6, and the results from the battles between “UofA” and “g4b” are in Figure 7.

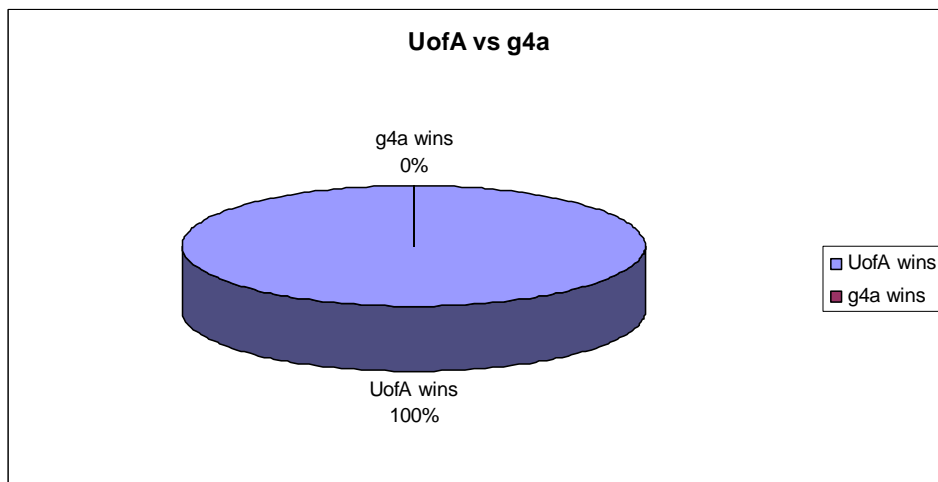


Figure 6. The results from the battles between “UofA” and “g4a”

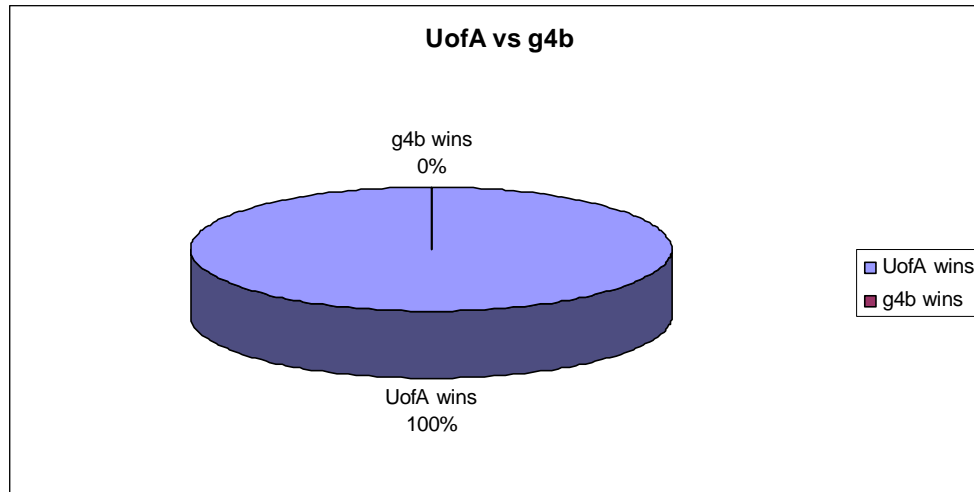


Figure 7. The results from the battles between “UofA” and “g4b”

Figure 6 and Figure 7 shows the total dominance of the AI design “UofA.” In the beginning, these results were a little disappointing, but in fact, this situation was very helpful. There are three main reasons why these results are important.

First, the design of “UofA” demonstrates complex behaviors and an AI solution must incorporate an advanced wining strategy to be competitive with it. Second, without existence of “UofA,” the design of “g4b” could be classified as “the best,” which obviously was not true. Third, the battles with “UofA” reveal specific weaknesses of the test cases, and knowledge of these weaknesses is the main source for improvement.

The next step of the research was to conduct a deeper analysis of competition results. At this stage, there were two key sources of information: the set of MOEs and the built-in capability of ORTS for 2D visualization.

The MOE percentage of wins demonstrates the dominance of the design for “g4b” over “Circle” and “g4a,” but it does not have the power to show why, nor why “g4a” and “g4b” perform so badly against “UofA.” On the other hand, the other two MOEs explain what the possible problems are. The results presenting these MOEs are shown in Figure 8, Figure 9, Figure 10, and Figure 11. Figure 8 and Figure 9 represent the distribution of the duration of each battle in the battles against “UofA.”

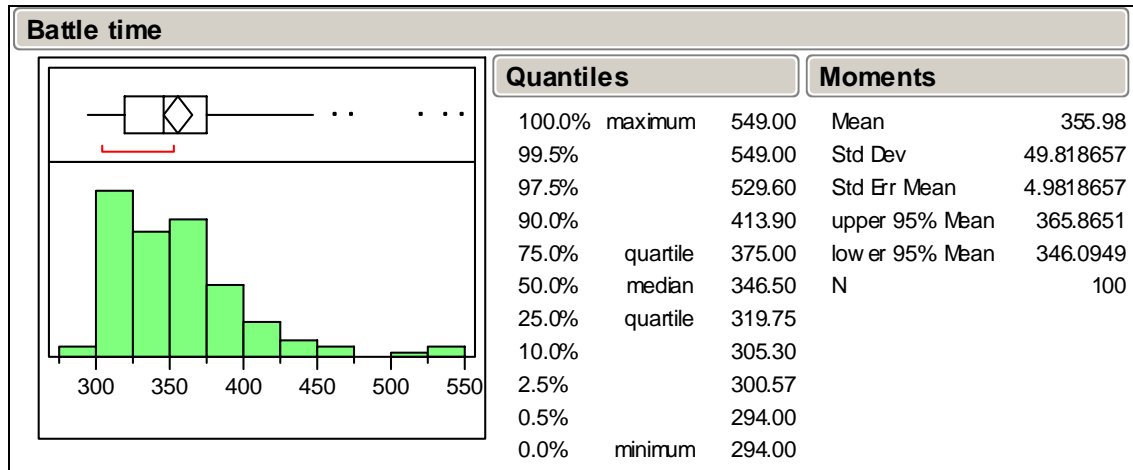


Figure 8. Distribution of the time of battle in the battle between “UofA” and “g4a”

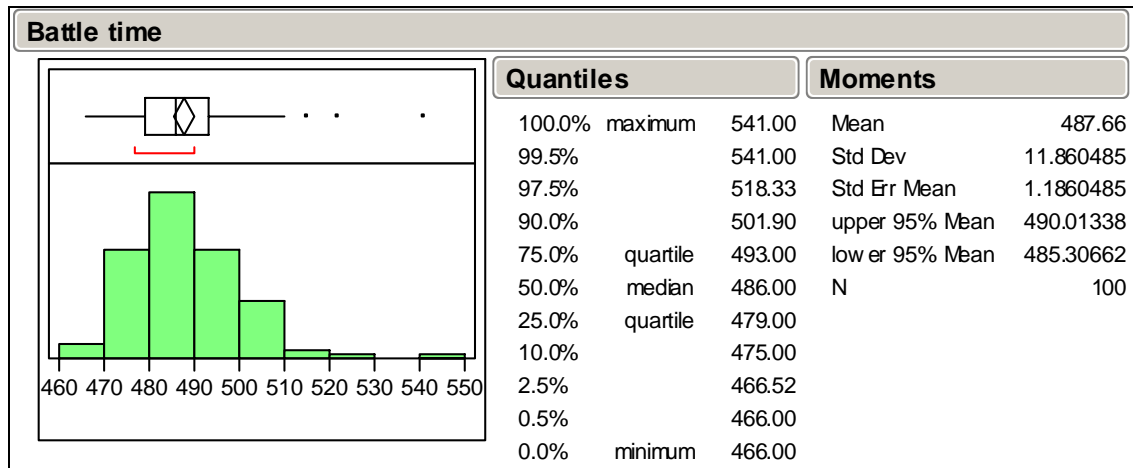


Figure 9. Distribution of the time of battle in the battle between “UofA” and “g4b”

These figures show that the battles between “UofA” and “g4a” take on average 355.98 seconds. In comparison, the battles between “UofA” and “g4b” have on average of 487.66 seconds. This shows clearly that “g4b” has better staying power than “g4a.” At the same time, the standard deviation of the duration of the battle for “g4a” is 49.82 seconds versus 11.86 seconds for “g4b,” which can be an indicator that the tactic of “g4a” is more unpredictable.

Figure 10 and Figure 11 represent the distribution of the casualties of the “marines” controlled by “UofA” at the end of corresponding battles.

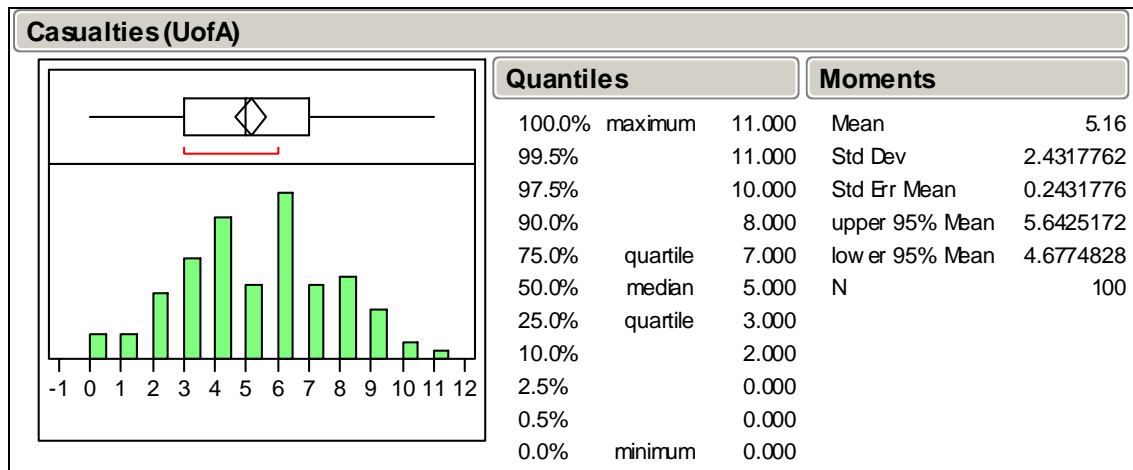


Figure 10. Distribution of the casualties of “UofA” in the battle between “UofA” and “g4a”

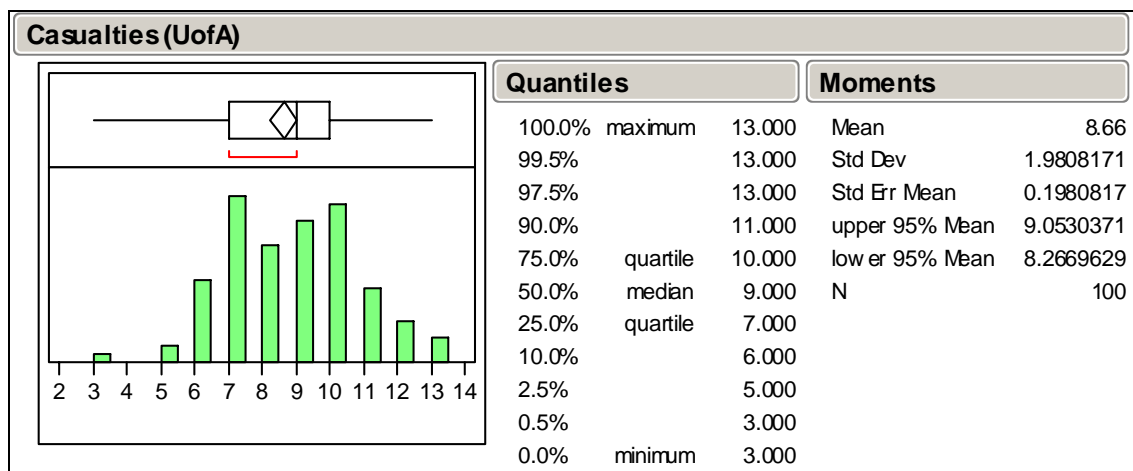


Figure 11. Distribution of the casualties of “UofA” in the battle between “UofA” and “g4b”

By comparing these two figures, it is clear that “g4b” has better lethality. On average at the end of the battle, “g4b” managed to eliminate 8.66 soldiers against 5.16 eliminated soldiers by “g4a.”

Using the built-in visualization capabilities in ORTS, Figure 12 and Figure 13 were produced. Figure 12 presents a snapshot of a battle between the designs of “UofA” and “g4a.” The “marines” controlled by the “UofA” are on the left side and the “marines” controlled by “g4a” are on the right side.

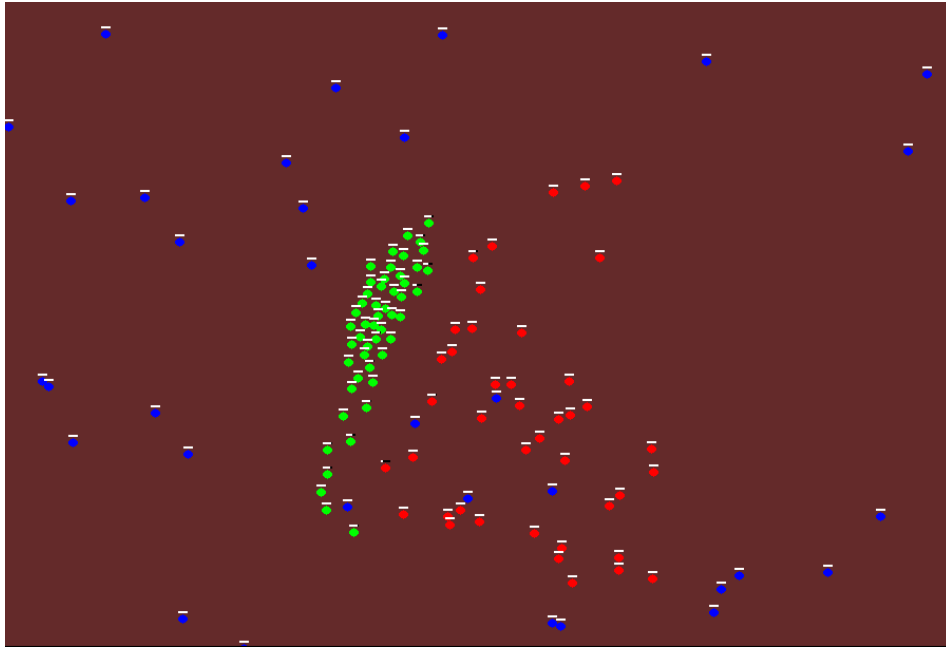


Figure 12. A battle between “UofA” and “g4a” designs

This shows the main problem with the AI design for “g4a:” the forces are too dispersed on the battlefield. They are facing the opponent’s forces one by one, and the overall concentration of forces is missing. At the same time this dispersion does not allow “UofA” to make a proper flanking formation, which is confirmed by the tail of the distribution for the duration of the battle on Figure 8.

Figure 13 presents a snapshot of a battle between the “UofA” and “g4b” designs. The “marines” controlled by the “UofA” are on the right side and the “marines” controlled by “g4b” are on the left side.

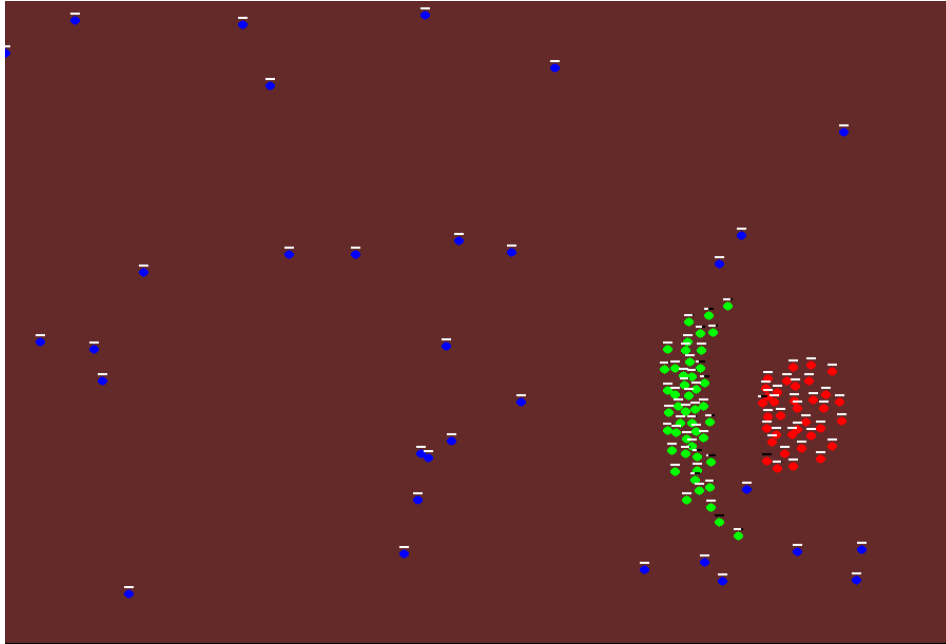


Figure 13. A battle between “UofA” and “g4b” designs

Figure 13 demonstrates that the “marines” controlled by “g4b” are in a good condensed formation, approaching the opponent’s forces. The problem is that not all of them are in a position to fire. Only the left edge of the formation is facing the opponent forces, and the right edge is out of fire range. This means, that despite the fact that the “marines” of “g4b” are in condensed formation, their firing power is dispersed. Therefore, once again Lanchester’s principle of force concentration is missing. This lack of force concentration is not as bad as in the design of “g4a,” but still it is the main factor for poor performance against “UofA.”

B. MODIFICATIONS IN THE AI DESIGN

The next part of this research was to try to improve the test AI designs. The decision was to build a new AI design using as a base “g4b.” The codename for this new design was “g4c.” The design of “g4b” was force driven, so the question was, can the driving forces be modified so that “g4c” can obtain a winning strategy?

The analysis for the results from the experimental competition demonstrates three properties of AI designs that have the potential to lead to winning strategy. First, the

“marines” should face the opponent’s forces in condensed formation. Second, this formation must be flexible enough to position the maximum number of “marines” in firing range. Third, the tactics of “marines” should include outflanking maneuvers demonstrated by “UofA.”

Translation of these desired properties to force manipulation is not a trivial task. Usually, the reverse tasks are solved in CAS simulations. The agents are influenced by a set of predefined forces, and in reaction to these forces, the agents demonstrate emergent behavior. In other words, the set of forces is known and the behavior is unknown. In this case, the task is opposite, the desired behavior is known, but the set of forces leading to this behavior are unknown.

A solution to this problem is by a trial and error approach, but because combat is such a complex problem, this can take a large number of iterations before the correct combination is discovered. The alternative approach is to check how each force is contributing to a “marine’s” behavior, and to try to find the right mixture. Moreover, it is not enough to produce the correct mixture of forces, but it is important to determine in what moment the “marines” will react to these forces.

To address these problems in the design of “g4c,” the behavior of “marines” was separated into three stages:

- First, “marines” must make a condensed formation.
- Second, when the “marines” are in the range of fire, they must start firing, and conduct outflanking tactics.
- Third, if the opponent’s forces are out of firing range, the “marines” must pursue them.

Applying these three stages to the behavior demonstrated by the “marines” in the “g4b” design, it was clear that for the first and the last stage there was no need to change the tactics. The problem was the second stage and the modifications were focused on that stage.

The new suggested set of forces influencing the “marine’s” behavior in the stage of exchange of fire is shown in Figure 14.

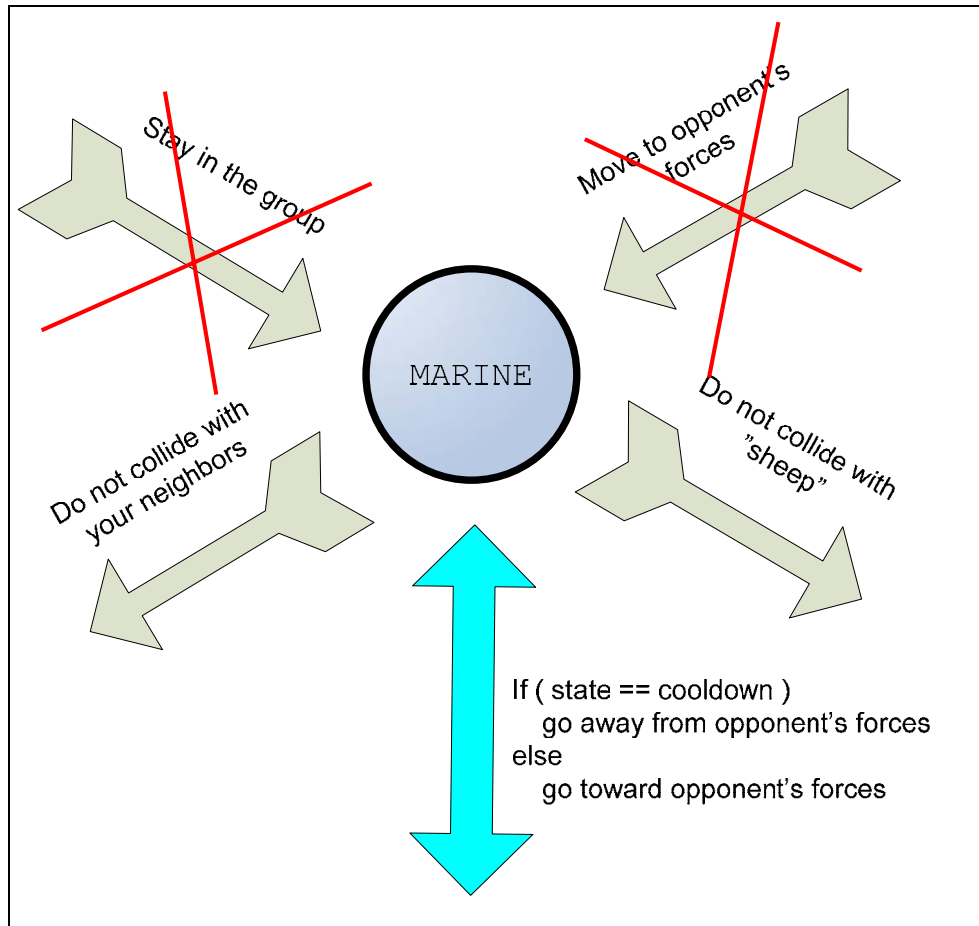


Figure 14. Forces influencing a “marine’s” behavior for the “g4c” design

Figure 14 is basically a repetition of Figure 3. The difference is that the force “Move to opponent’s forces” was replaced with a new force, the direction of which depends on the state of the “marine’s” weapon. At the same time, the force “Stay in the group” also was eliminated, which allowed “marines” to make more dispersed outflanking formations.

The conditions for switching between different behaviors were:

- After the ORTS game starts, for a certain amount of time, the “marines” are reacting to the forces shown in Figure 3.
- When the “marines” are in a firing range with the opponent’s formation, they start to react to the forces shown in Figure 14.
- Finally, if the opponent’s forces retreat, and are out of firing range, again the “marines” will start to react to the forces shown in Figure 3.

C. RESULTS FROM THE SECOND RUN OF THE EXPERIMENTAL COMPETITION

The experimental competition was executed again for test cases “g4c” and the base cases “Circle” and “UofA.” The results for the main MOE, the percentage of wins, from the battles between “Circle” and “g4c” are presented in Figure 15.

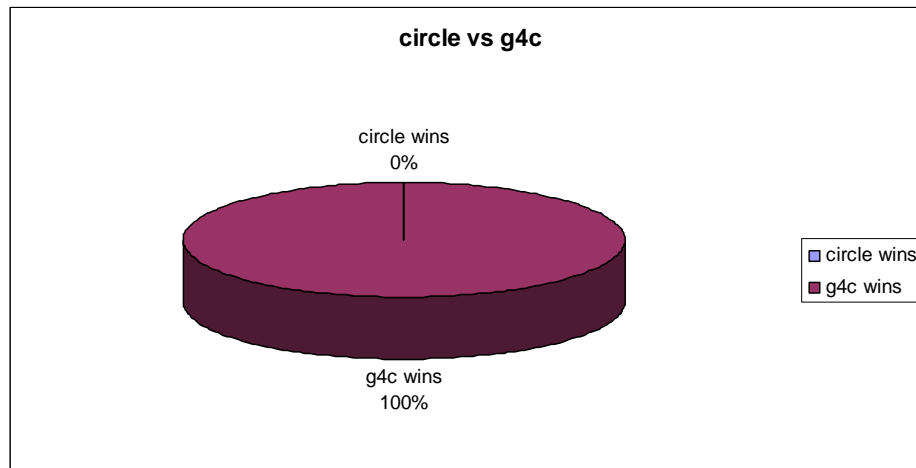


Figure 15. The results from the battles between “Circle” and “g4c”

As expected, because “g4c” is a continuation of “g4b,” it has total dominance over the design of “Circle.” The results for the same MOE, from the battles between “UofA” and “g4c” are presented in Figure 16.

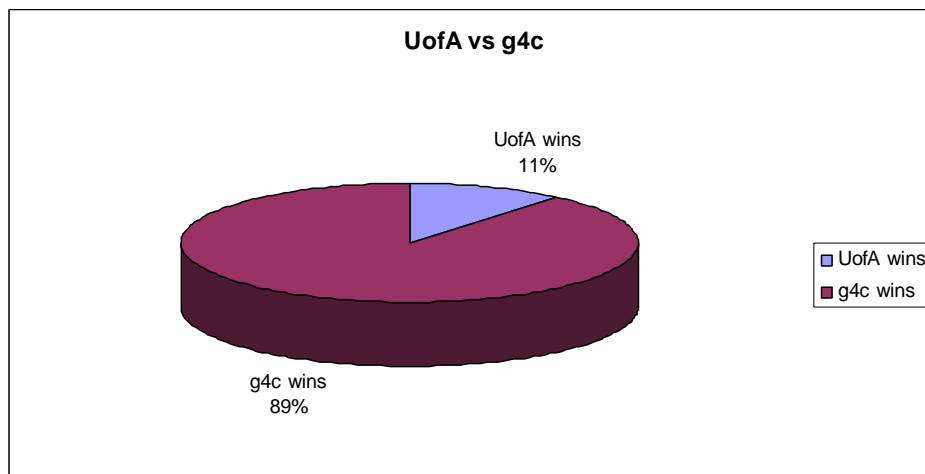


Figure 16. The results from the battles between “UofA” and “g4c”

The results demonstrate significant improvement in the tactics shown by “g4c” in comparison to “g4b.” The “marines” controlled by the design of “g4c” managed to win almost ninety percent the battles. The main reason is the superiority of the new “marines” formation, which is represented in a snapshot of a battle between “UofA” and “g4c” designs as shown on Figure 17. In Figure 17, the “marines” controlled by the “UofA” are on the left side and the “marines” controlled by “g4c” are on the right side.

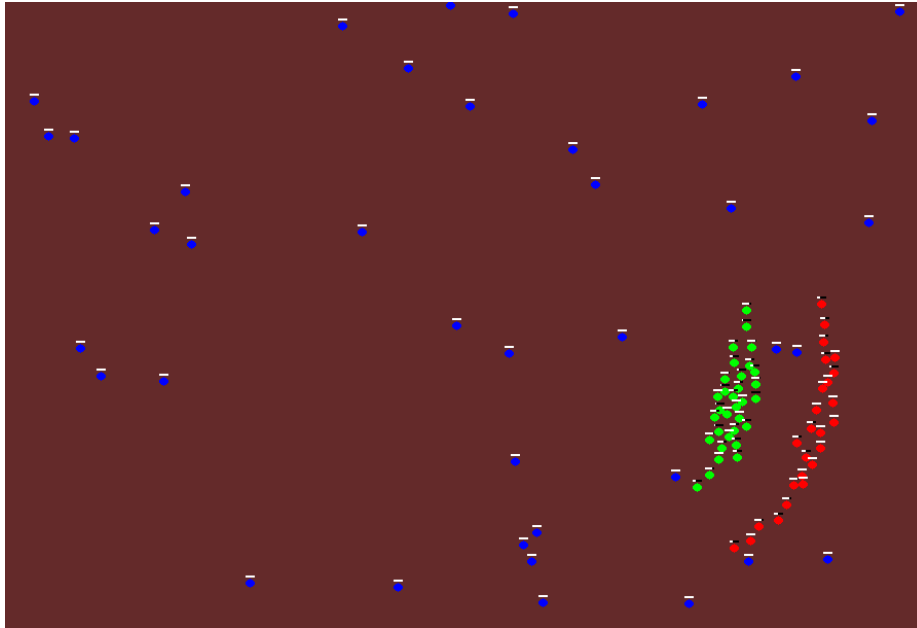


Figure 17. A battle between “UofA” and “g4c” designs

Figure 17 shows that “g4c” had a better outflanking strategy than “UofA.” This strategy is better because the design of “g4c” managed to put more “marines” in the front edge of the formation. In some cases, “g4c” even achieved putting all the “marines” in a curve surrounding the opponent’s formation within firing distance. This is the best solution from the perspective of force concentration. This allows the “marines” to advance, fire, and then wait for the weapon’s cooldown at a safe distance. This increases “marines” survivability, which is indicated in Figure 18.

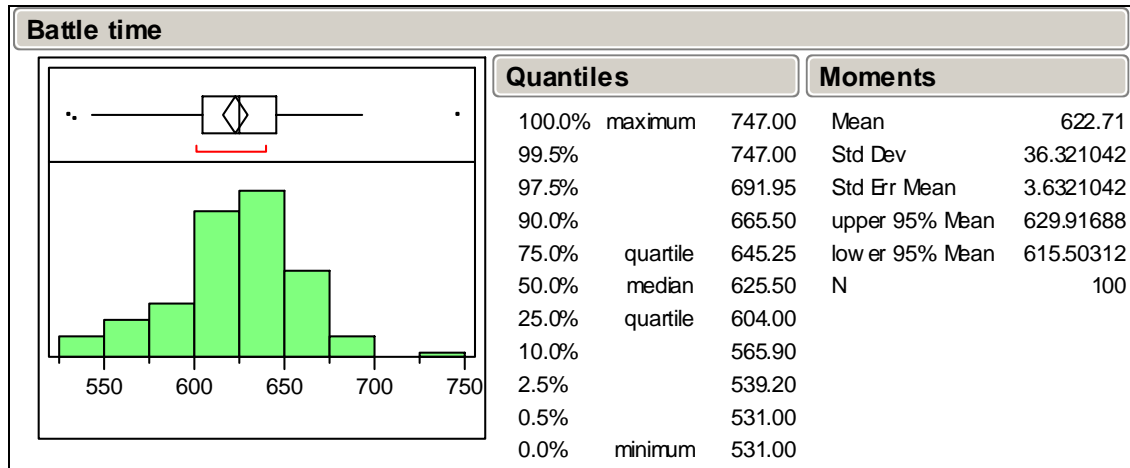


Figure 18. Distribution of the time of battle in the battle between “UofA” and “g4c”

Figure 18 represents the distribution of the MOE for the duration of the battle. It shows that on average the battles lasts for 622.71 seconds longer than the 487.66 seconds registered in the battles between “UofA” and “g4b.” This is clear evidence of the increased staying power of “g4c.” At the same time, the next MOE for casualties at the end of the battle reveals some additional information about the design of “g4c.” The distributions of the MOE for opponent casualties at the end of the battle between “UofA” and “g4c” are presented in Figure 19 and Figure 20.

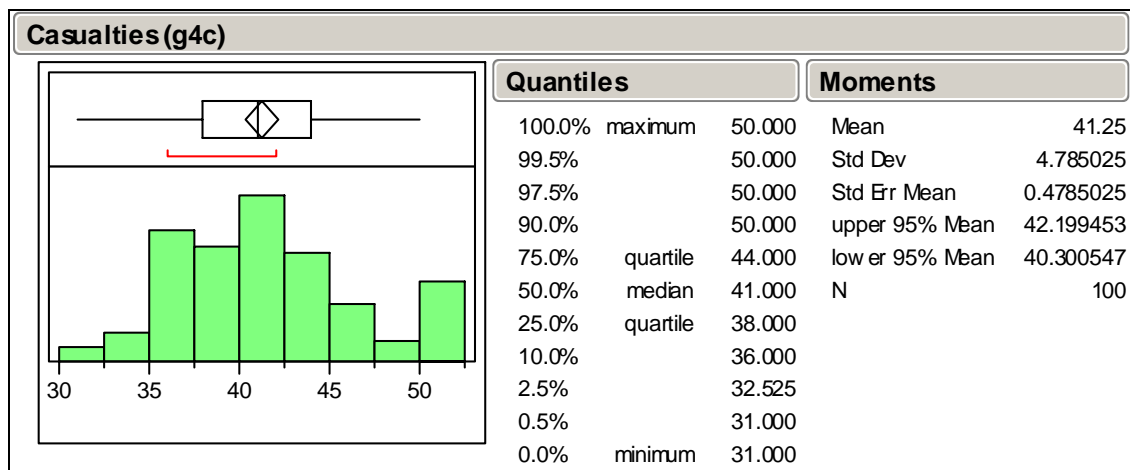


Figure 19. Distribution of “g4c” casualties in the battles between “UofA” and “g4c”

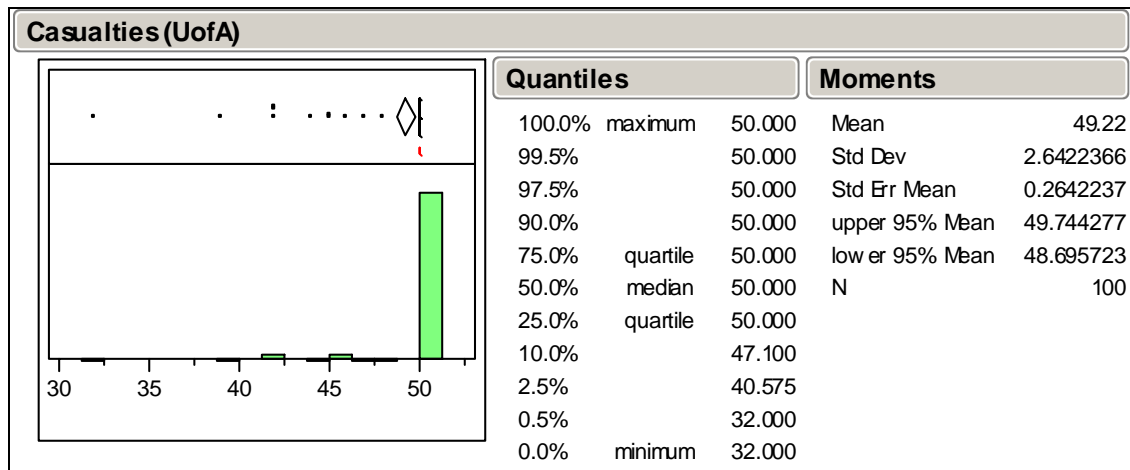


Figure 20. Distribution of “UofA” casualties in the battles between “UofA” and “g4c”

From Figure 19, it is interesting to note that on average the casualties of “g4c” at the end of the battle are 41.25 soldiers. This means that the battle is very tight and “g4c” wins by approximately eight “marines” on average. The fact that the advantage of “g4c” is so tiny is supported by the standard deviation of the distribution, which is 4.78.

Analyzing Figure 20, the big bin of fifty casualties at the end of the battle corresponds to the losses of “UofA,” in which the entire set of 50 “UofA” “marines” is eliminated, but the few wins that “UofA” has are in the range of 32 to 47, which again demonstrates the tightness of the battle.

In conclusion, the results show that the design of “g4c” has evolved by employing tactics similar to the complex tactics used by “UofA,” but at the same time, it has achieved a slight advantage.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

In the post 9/11 world, dominated by asymmetric warfare and operations other than war, there is evidence of a trend of increasing usage of AI solutions in military simulations and beyond. In the U.S. Department of Defense, AI solutions are used not only for military simulations, but also for training systems and autonomous vehicles. It is believed that AI solutions will play a major role in future network centric combat. This emphasizes the importance of research in the AI domain for military applications, and the thesis contributes to this growing research.

Overall, the goals of the thesis research are achieved. The thesis research succeeded in the development of a framework and methodology to apply the competitive approach. In addition, the results presented in the thesis demonstrated the advantages of the competitive approach in AI designs used in military simulations. These advantages can be summarized in the following aspects. First, the ORTS engine was tested and it was found to be a suitable platform for exploring new AI designs with application to military simulations. Second, the research results show that the competitive approach leads to improvements of AI designs for ORTS, and this can reasonably be expected to extend to those used in military simulations. Third, and most important, the advantage of the competitive approach for AI designs is that it gives valuable inside information about the problem, which is almost impossible to obtain otherwise.

This research clearly implies that the competitive approach has more universal application. It can be used not only for battles between two AI designs, but also for competitions between AI designs of any kind. For example, in ORTS game one, the main goal is to collect “minerals” in a certain time. The competition here is testing which AI design would do a better job by collecting more minerals in that time. This small example shows that the competitive approach has great potential for improving not only

military simulation, but also AI designs for other problem domains, such as development of more advanced algorithms for driving autonomous vehicles in collaborative environments.

B. RECOMMENDATIONS FOR FUTURE WORK

Analyzing the results from the second run of the experiment, some remarks about future improvement of the AI design for “g4c” should be made.

First, despite the fact that the design of “g4c” demonstrates the advanced behavior, there is still room for improvement. For example, the “g4c” “marines” make outflanking formations as a reaction of the advance of the opponent’s forces. It would be better if they could create this outflanking formation before meeting the opponent’s forces.

Second, as was stated, in this thesis only the spatial part of the strategy for the ORTS game four was considered, and the fire allocation part was solved by using heuristic algorithms. Although these heuristic algorithms have good performance, they did not provide the best possible solution. The winning tactic is a complex problem, and ignoring a major part of it does not lead to developing an AI design that is close to optimal. In highly developed AI designs such as “UofA” and “g4c” every advantage, even the smallest one, can be influential to the outcome of the competition, as seen in Figure 19 and Figure 20. As a next step of research, it is important to explore how a better fire allocation algorithm would affect the AI design performance. Because the nature of warfare is nonlinear, this must be done in an integrated way, advancing the movement part and the fire allocation part together.

In addition to improving the design of “g4c,” the ORTS engine, itself, needs improvements. By analyzing the report capabilities of the ORTS engine, it is clear that a more advanced tool that extracts the data from the simulation is needed. This tool must be able to replay a battle forward and backward with all corresponding data. In real-time environments it is important to be able to register every change in the state of the

simulation, and to replay it back in the context of the 2D or 3D environments. At the same time, this tool should not affect the results of the simulation, which in the framework of real-time simulations is a difficult task.

Another direction of research is implementing the competitive approach in military-oriented simulation engines such as SimKit and Delta 3D. This might take the form of a separate project combining the ideas from military simulations, virtual reality, RTS games, FPS games, and the tournament manager.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

- 2007 ORTS RTS Game AI Competition. Retrieved March 2008 from <http://www.cs.ualberta.ca/~mburo/orts/AIIDE07/index.html>.
- Adams, D. (2006). The state of the RTS. Retrieved February 2008 from <http://pc.ign.com/articles/700/700747p1.html>.
- Back, D. (2002). Agent-based soldier behavior in dynamic 3D virtual environments (Master's thesis, Naval Postgraduate School, 2002).
- Bitinas, E. J., Henscheid, Z. A., & Truong, L. V. (2003). Pythagoras: A new agent-based simulation system. *Technology Review Journal*, 45-58.
- Bracken, J., Kress, M., & Rosenthal, R. E. (1995). *Warfare modeling*, Military Operations Research Society. New York: John Wiley & Sons.
- Buro, M., & Furtak, T. (2003). RTS games as test-bed for real-time research. *Invited Paper at the Workshop on Game AI*, JCIS, 481-484.
- Buro, M., & Furtak, T. (2004). RTS games and real-time AI research. *Proceedings of the Behavior Representation in Modeling and Simulation Conference*, 34-41.
- Buro, M., & Furtak, T. (2005). On the development of a free RTS game engine. *GameOn'NA Conference*, 23-27.
- Buro, M. (2002). ORTS: A hack-free RTS game environment. *Proceedings of the International Computers and Games Conference*, 280-291.
- Buro, M. (2003). Real-time strategy games: A new AI research challenge. *Proceedings of the International Joint Conference on AI*, 1534-1536.
- Buro, M. (2004). Call for AI research in RTS games. *Proceedings of the AAAI-04 workshop on AI in games*, 139-142.
- Buro, M., Bergsma, J., Deutscher, D., Furtak, T., Sailer, F., Tom, D., & Wiebe, N. (2006). AI system designs for the first RTS-game AI competition. *Proceedings of the GameOn Conference*, 13-17.
- Buss, A. (2002). Component based simulation modeling with Simkit. *Proceedings of the 2002 Winter Simulation Conference*, 243-249.
- Cox, C., & Fu, D. (2005). AI for automated combatants in a training application. *Proceedings of the second Australasian conference on interactive entertainment*, 57-64.

- Darken R., McDowell P., & Johnson E. (June 2005). The Delta3D open source game engine. *IEEE Computer graphics and applications*, 10-12.
- Devkar, A. (2003). Command and conquer in the development of real-time strategy. Retrieved February 2008 from <http://www-sul.stanford.edu/depts/hasrg/histsci/STS145papers/Devkar.htm>.
- Henderson, C., & Rodriguez, A. (2002). Modeling in OneSAF. *Proceedings of the 11th Conference on Computer Generated Forces and Behavior Representation*, 337-347.
- Henry, H. (1994). The corps battle simulation: Remodeling the model for new missions. *Proceedings of the 1994 Winter Simulation Conference*, 788-791.
- Hughes, W. P. (1995). A salvo model of warships in missile combat used to evaluate their staying power. *Naval Research Logistics*, 42(2), 267-289.
- Ilachinski, A. (2000). Irreducible semi-autonomous adaptive combat (ISAAC): An artificial-life approach to land combat. *Military Operations Research*, 5(3), 29-46.
- Keefer, J. A good RTS? Give me gameplay and give me death! Retrieved February 2008 from http://archive.gamespy.com/legacy/articles/rtsmusthave_a.shtm.
- Keefer, J. The evolution of the PC real-time strategy game. Retrieved February 2008 from http://archive.gamespy.com/legacy/articles/rts history_a.shtm.
- Keefer, J. Top 10 RTS innovations. Retrieved February 2008 from <http://archive.gamespy.com/top10/january01/rtsinnovation/index.shtm>.
- Lauren, M., & Stephen, R. (2002). Mana: Map-aware non-uniform automata: A New Zealand approach to scenario modeling. *Journal of Battlefield Technology* 5, 27-31.
- Lauren, M. (1999). Characterising the difference between complex adaptive and conventional combat models. Defence Operational Technology Support Establishment, Auckland, New Zealand.
- Lenoir, T., & Lowood, H. (2002). Theaters of war: The military-entertainment complex. Stanford University. Retrieved February 2008 from http://www.stanford.edu/class/sts145/Library/Lenoir-Lowood_TheatersOfWar.pdf.
- Lucas, T. W., & Turkes, T. (2003). Fitting Lanchester Equations to the Battles of Kursk and Ardennes. *Naval Research Logistics*, 51(1), 95-116.
- Miller, J. H., & Page, S. E. (2007). *Complex adaptive systems: An introduction to computational models of social life*. Princeton, NJ: Princeton University Press.

- MITRE Corp. (January 2003). Use of free and open-source software (FOSS) in the U.S. Department of Defense, ver. 1.2.04. Retrieved March 2008 from http://www.terrybollinger.com/dodfoss/dodfoss_pdf.pdf.
- Moffat, J. (September 2003). Complexity theory and network centric warfare US DoD command and control research program. Retrieved March 2008 from http://www.dodccrp.org/files/Moffat_Complexity.pdf.
- Mueller, A. (2006). Survey of available artificial intelligence technologies for addition into Delta3D (Master's thesis, Naval Postgraduate School, 2006).
- Pawloski, J. (2001). Modeling tactical level combat using a multi-agent system design paradigm (GI Agent) (Master's thesis, Naval Postgraduate School, 2001).
- Posadas, S. (2001). Stochastic simulation of a Commander's Decision Cycle (SSIM Code) (Master's thesis, Naval Postgraduate School, 2001).
- Reece, D. A., Kraus, M., & Dumanoir, P. (2000). Tactical movement planning for individual combatants. *9th Conference on Computer Generated Forces and Behavioral Representation*. Retrieved November 2007 from <http://www.sisostds.org/index.php?tg=fileman&idx=get&id=2&gr=Y&path=CGF-BR%2F9th+CGF-BR%2F9TH+Abstracts%2C+Papers+and+Presentations&file=9TH-CGF-045.doc>.
- Reynolds, C. W. (1987). Flocks, herds, and schools: A distributed behavioral model, in Computer Graphics, *SIGGRAPH '87 Conference Proceedings*, 21(4), 25-34.
- Russell, J., & Norvig, P. (2003). *Artificial intelligence: A modern approach* (2nd ed.). Upper Saddle River: Prentice Hall/Pearson Education.
- Taylor, J. G. (1983). Lanchester models of warfare. Operations Research Society of America, Military Applications Section, Arlington, VA.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California
3. Atanas Nachev
Defence Advanced Research Institute
Rakovski Defence and Staff College
Sofia, Bulgaria